

LVMAS

Das mobile Agentensystem
in NI LabVIEW

Frederik Berck
GSI Darmstadt
f.berck@gsi.de

Übersicht

- Das mobile Agentensystem im Allgemeinen
- LVOOP und die HGF Basisklassen
- LVMAS
- Ende



Was ist ein mobiles Agentensystem?

Was ist ein Agent?

- „An agent is a program that assist people and acts on their behalf. Agents function by allowing people to delegate work to them.“
- „Programming and Deploying JAVA Mobile Agents with Aglets
- Danny B. Lange & Mitsuru Oshima

Anforderungen an einen Agenten

- Autonomie
- Zielgerichtetes Handeln
- Reaktiv
- Ununterbrochene Ausführung

Weitere mögliche Eigenschaften eines Agenten

- Mobil
- Kommunikativ
- Adaptiv
- Glaubhaft

Wann ist ein Agent mobil?

- Voraussetzung: Netzwerk mit mehr als einem Host
- Der Agent kann seine Arbeit unterbrechen, und mitsamt seinen Eigenschaften und Funktionen auf einen anderen Host wechseln, auf dem er anschließend seine Arbeit fortsetzt

Was ist ein mobiles Agentensystem

- Ein Komplettpaket, das ein Netzwerk schafft, in dem verschiedene Hosts Arbeitsumgebungen bereitstellen, in denen verschiedene Agenten ihren individuellen Aufgaben nachgehen können.

Übersicht

- Das mobile Agentensystem im Allgemeinen
- LVOOP und die HGF Basisklassen
- LVMAS
- Ende

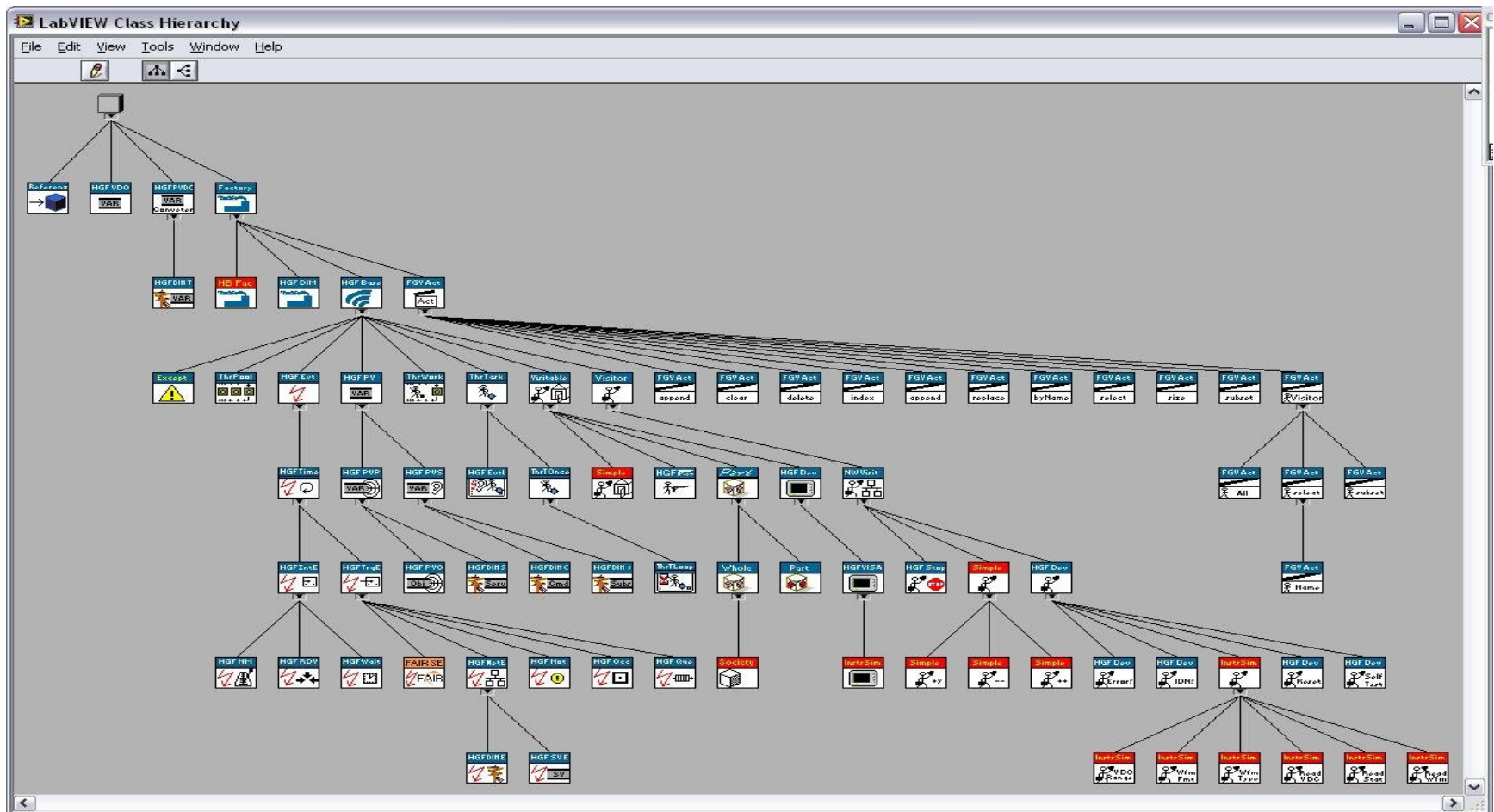
Grundlegende Unterschiede zwischen Java und LabVIEW

Java	LabVIEW
Textbasierte Programmierung	Grafische Programmierung
Objekte und Methoden sind gekoppelt	Objekte repräsentieren passive Daten
Objekte werden durch Referenzen dargestellt und sind somit Entitäten	Objekte sind Daten, die bei Leitungsabzweigen kopiert werden
Programmablauf über Function Calls	Datenfluss

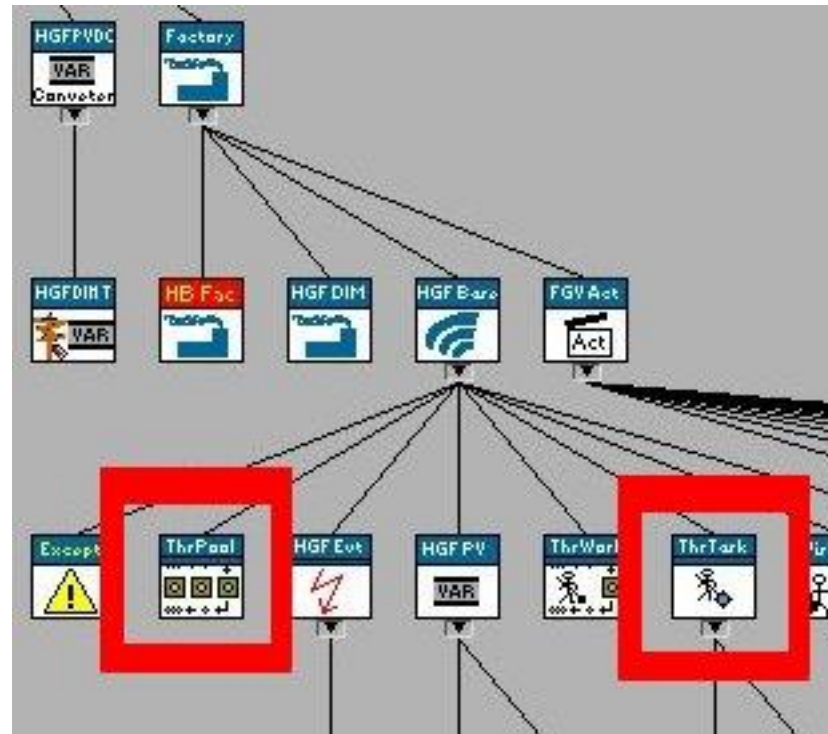
Realisierung mittels LVOOP

- Als Basis für die Realisierung des mobilen Agentensystems dient LVOOP, die **LabVIEW objektorientierte Programmierung. Mit den HGF Basisklassen wurde schon der Grundstein für das Agentensystem bereitgestellt.**

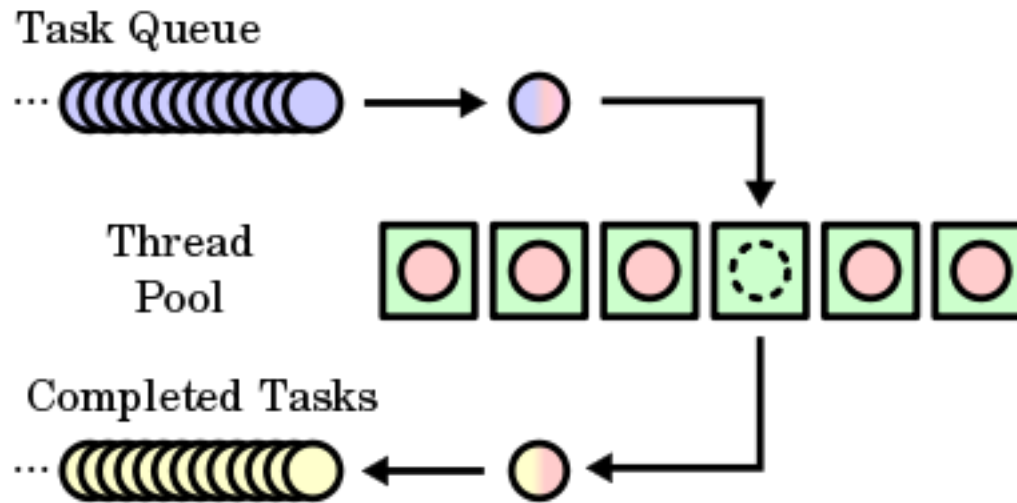
Die HGF Basisklassen



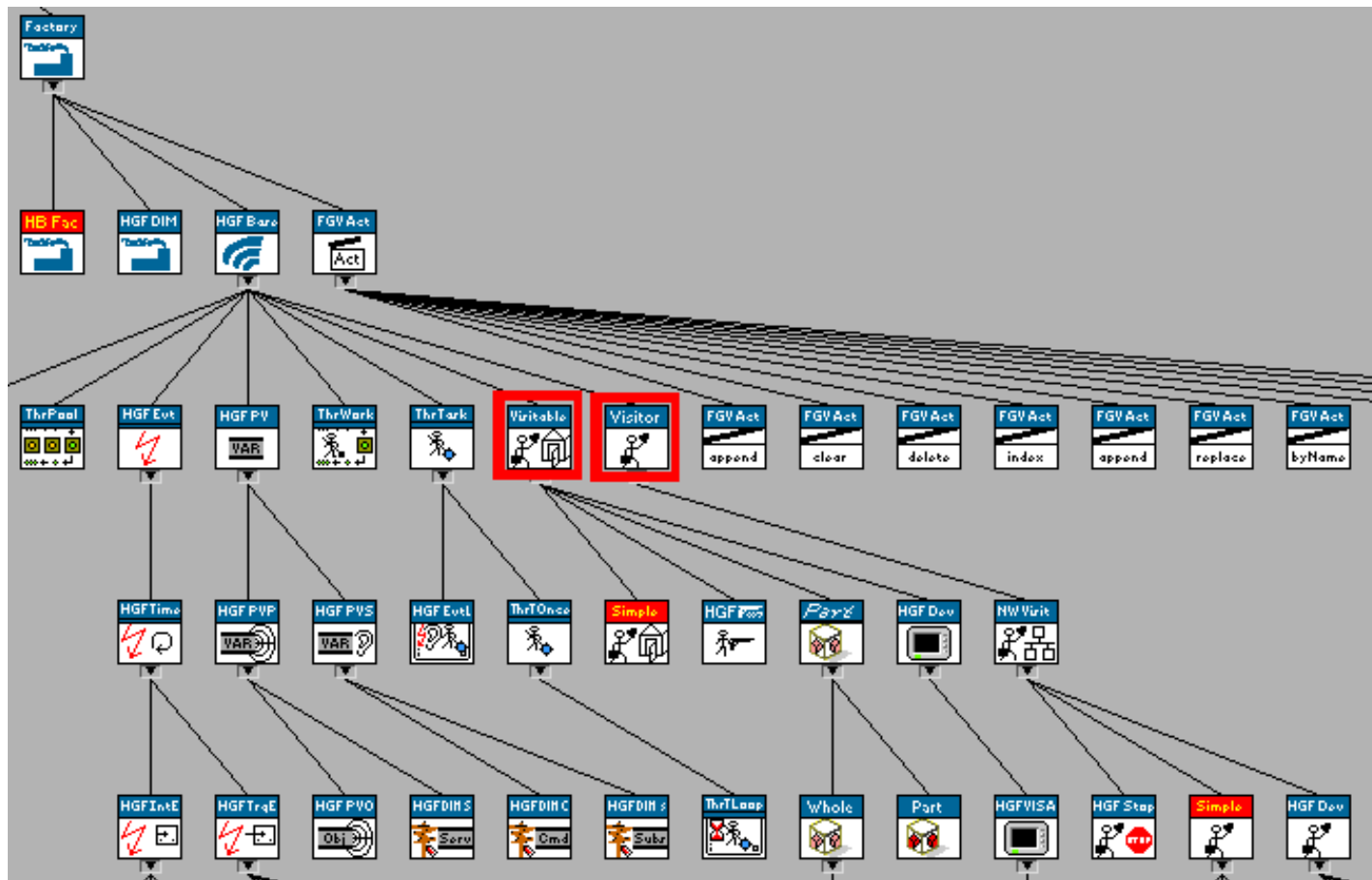
Threadpool & Task



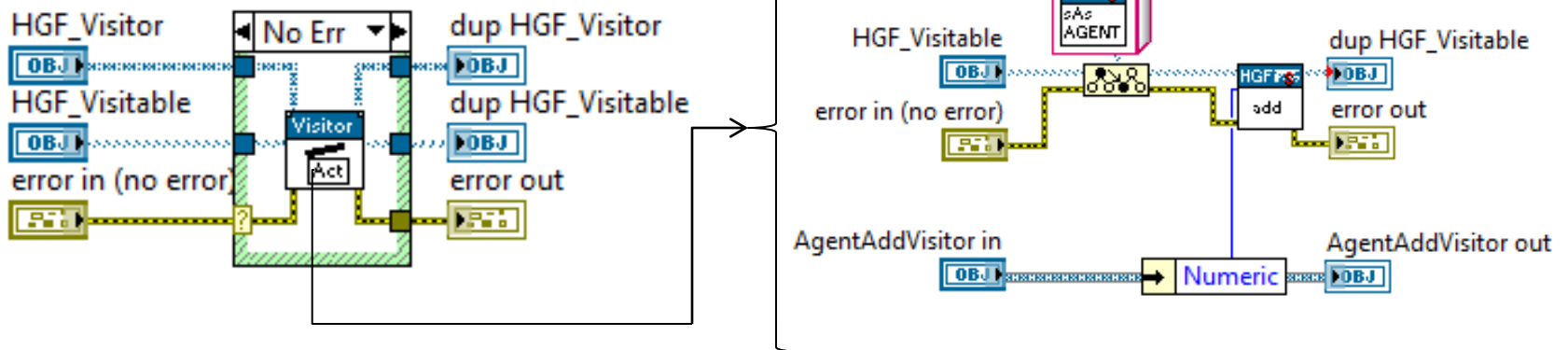
Threadpool & Task



Visitable & Visitor



Visitable & Visitor



Übersicht

- Das mobile Agentensystem im Allgemeinen
- LVOOP und die HGF Basisklassen
- LVMAS
- Ende

LVMAS

Das LabVIEW Mobile Agent System

Visitable „Mobile Agent“

- Hat von Visitable geerbt
- Passives Objekt

Task „Mobile Agent State Machine“

- Kann als Motor für den Agenten aufgefasst werden
- Ein Task, der eine Zustandsmaschine beinhaltet
- Jeder Threadpool kann eine bestimmte Anzahl an Tasks parallel aufnehmen und ausführen

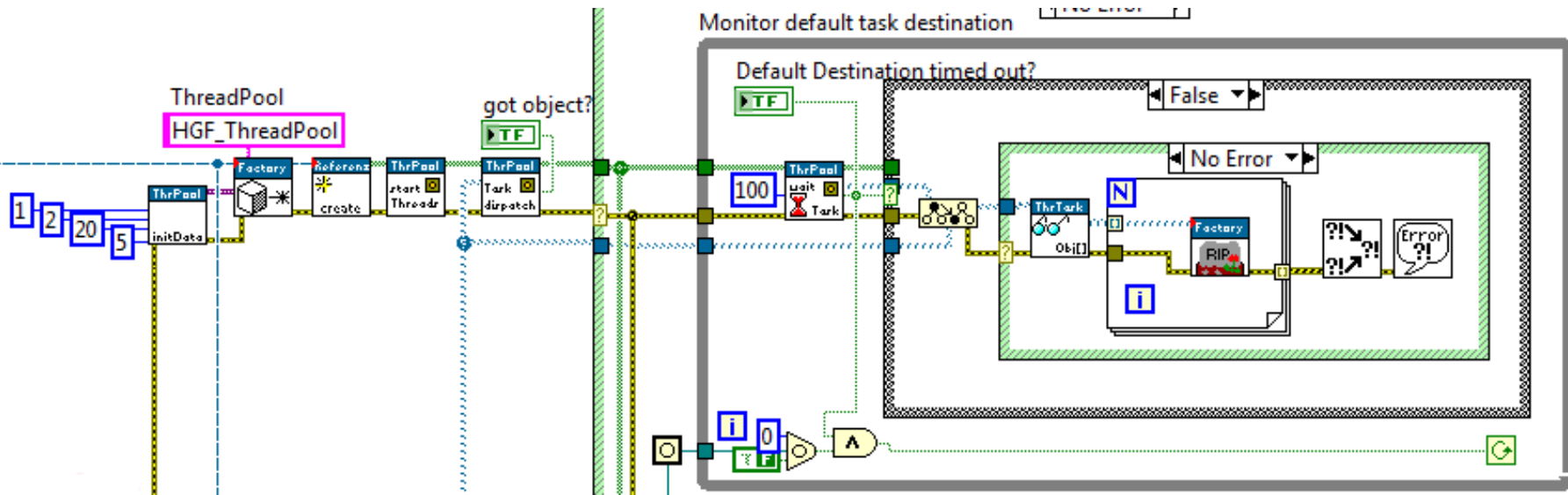
Die Statemachine

- Zustandsmaschine, die **ein** Agentenobjekt in den internen Daten aufnehmen kann
- Visitoren können durch den Task an die Statemachine übergeben werden
- Zuständig für:
 - Initialisierung der Agenten (Anlegen von Schnittstellen)
 - Überprüfung von Visitoren
 - Zustellung der Visitoren zum Agentenobjekt

Der LabVIEW Agent

- Agentenklasse als passives Objekt
- Zustandsmaschinen-Task als Engine für den Agenten
- Kombination aus beidem entspricht dem ursprünglichen Bild eines Softwareagenten

Threadpool



Host

- Executable oder Start-VI
- Startet eine Verwaltung, einen stationären Host-Agenten
- Stellt einen Threadpool bereit
- Reentrant – mehrere Hosts auf einem Rechner

„Host-Agent“

- Stationärer Agent
- Stellt die Arbeitsumgebungen für Agenten bereit
- Startet für ankommende Agenten einen Task im zuständigen Thread Pool
- Zuständig für Sicherheit

Wie wird ein LabVIEW-Agent mobil

- Die Daten eines Agenten befinden sich innerhalb des Agentenobjektes, welches passiv ist. Überträgt man dieses Objekt, so kann der Agent an einem anderen Ort mit seiner Tätigkeit fortfahren. LabVIEW bietet die Möglichkeit ein komplettes Objekt mitsamt seinen Attributen in einen String zu wandeln (Flatten to String). Diesen kann man über verschiedene Wege an einen anderen Host übertragen und rückwandeln (Unflatten from String). Hier muss noch eine Zustandsmaschine mit diesem Objekt gestartet werden.

ENDE

Vielen Dank für Ihre Aufmerksamkeit